AMENDMENTS TO THE SPECIFICATIONS:

Please replace paragraph [0001] with the following amended paragraph:

[0001] This application claims priority to U.S. Provisional Application Serial No. 60/400,391 titled "JSM Protection," filed July 31, 2002, incorporated herein by reference. This application also claims priority to EPO Application No. 03291927.6, filed July 30, 2003 and entitled "A Multi-Processor Computing System Having A Java Stack Machine And A RISC-Based Processor," incorporated herein by reference. This application also may contain subject matter that may relate to the following commonly assigned co-pending applications incorporated herein by reference: "System And Method To Automatically Stack And Unstack Java Local Variables," Serial No. 10/632,228, filed July 31, 2003~~, Attorney Docket No. TI-35422 (1962-05401)~~; "Memory Management Of Local Variables," Serial No. 10/632,067, filed July 31, 2003~~, Attorney Docket No. TI-35423 (1962-05402)~~; "Memory Management Of Local Variables Upon A Change Of Context," Serial No. 10/632,076, filed July 31, 2003~~, Attorney Docket No. TI-35424 (1962-05403)~~; "A Processor With A Split Stack," Serial No. 10/632,079, filed July 31, 2003~~, Attorney Docket No. TI-35425(1962-05404)~~; "Using IMPDEP2 For System Commands Related To Java Accelerator Hardware," Serial No. 10/632,069, filed July 31, 2003~~, Attorney Docket No. TI-35426 (1962-05405)~~; "Test With Immediate And Skip Processor Instruction," Serial No. 10/632,214, filed July 31, 2003~~, Attorney Docket No. TI-35427 (1962-05406)~~; "Test And Skip Processor Instruction Having At Least One Register Operand," Serial No. 10/632,084, filed July 31, 2003~~, Attorney Docket No. TI-35248 (1962-05407)~~; "Synchronizing Stack Storage," Serial No. 10/631,422, filed July 31, 2003~~, Attorney Docket No. TI-35429 (1962-05408)~~; "Methods And Apparatuses For Managing Memory," Serial No. 10/631,252, filed July 31, 2003~~, Attorney Docket No. TI-35430 (1962-05409)~~; "Write Back Policy For Memory," Serial No. 10/631,185, filed July 31, 2003~~, Attorney Docket No. TI-35431 (1962-05410)~~; "Methods And Apparatuses For Managing Memory," Serial No. 10/631,205, filed July 31, 2003~~, Attorney Docket No. TI-35432 (1962-05411)~~; "Mixed Stack-Based RISC Processor," Serial No. 10/631,308, filed July 31, 2003~~, Attorney Docket No. TI-35433 (1962-05412)~~; "Processor That Accommodates Multiple Instruction Sets And Multiple Decode Modes," Serial No. 10/631,246, filed July 31, 2003~~, Attorney Docket No. TI-35434 (1962-05413)~~; "System To Dispatch Several Instructions On Available Hardware Resources," Serial No. 10/631,585, filed July 31, 2003~~,~~

~~Attorney Docket No. TI-35444 (1962-05414)~~; "Micro-Sequence Execution In A Processor," Serial No. 10/632,216, filed July 31, 2003~~, Attorney Docket No. TI-35445 (1962-05415)~~; "Program Counter Adjustment Based On The Detection Of An Instruction Prefix," Serial No. 10/632,222, filed July 31, 2003~~, Attorney Docket No. TI-35452 (1962-05416)~~; "Reformat Logic To Translate Between A Virtual Address And A Compressed Physical Address," Serial No. 10/632,215, filed July 31, 2003~~, Attorney Docket No. TI-35460 (1962-05417)~~; "Synchronization Of Processor States," Serial No. 10/632,024, filed July 31, 2003~~, Attorney Docket No. TI-35461 (1962-05418)~~; "Conditional Garbage Based On Monitoring To Improve Real Time Performance," Serial No. 10/631,195, filed July 31, 2003~~, Attorney Docket No. TI-35485 (1962-05419)~~; "Inter-Processor Control," Serial No. 10/631,120, filed July 31, 2003~~, Attorney Docket No. TI-35486 (1962-05420)~~; "Cache Coherency In A Multi-Processor System," Serial No. 10/632,229, filed July 31, 2003~~, Attorney Docket No. TI-35637 (1962-05421)~~; and ~~"Concurrent Task Execution In A Multi-Processor, Single Operating System Environment," Serial No. 10/632,077, filed July 31, 2003, Attorney Docket No. TI-35638 (1962-05422).~~ "A Multi-Processor Computing System Having A Java Stack Machine And A RISC-Based Processor," Ser. No. 10/631,939, filed July 31, 2003.

Please replace paragraph [0037] with the following amended paragraph:

[0037] In at least some embodiments of the invention, the MPU **104** may prioritize multiple tasks when awake. For example, if the MPU **104** has been awoken by a system interrupt **209**, more than one interrupt source may have positioned the system interrupt **209** and the MPU **104** will perform the associated tasks according to their pre-determined priority. In at least some embodiments, an operating system ("O/S") running ~~of~~ on the MPU **104** may control the order in which the MPU **104** carries out multiple interrupt requests.

Please replace paragraph [0039] with the following amended paragraph:

[0039] In embodiments in which multiple signals (e.g. system interrupt **209**, system interrupt detect **216**, wait release **214**) occur simultaneously, approximately simultaneously, or concurrently, the operating system ("O/S") running on the MPU **104** may decide, according to a pre-determined priority, whether the MPU **104** will execute instructions as requested by the JSM **102** or execute the task(s) requested by the system interrupt **209**. As shown in ~~**FIG. 6**~~ **FIG. 3**,

the processor interrupt signal **218** asserted by the synchronization unit **206** as described above may be received by the MPU **104** as a system interrupt **209** and handled according to a pre-determined priority. More specifically, the embodiment described above enables the O/S running on the MPU **104** to schedule multiple types of support requests from the JSM **102** relative to their respective priority as compared to other requests (e.g., system interrupts **209**) handled by the MPU **104**. Some of the support request from the JSM **102** may have lower priority than some system interrupts **209**, while other support requests from the JSM **102** have a higher priority. The priority of a support request from the JSM **102** may be included in the status read by the MPU **104** as described above. The processor interrupt may itself contain the priority information as well.